In the Claims

Claims 1-24 (canceled).

25. (original) A method of condition generation comprising the steps of:

defining a set of arithmetic condition flags (ACFs);

determining side effects of a plurality of scalar conditions on an instruction by instruction basis;

setting a set of arithmetic scalar flags (ASFs) to save the determined side effects;

specifying a condition code utilizing a compare instruction; and

updating the ACFs based upon the specified condition code.

26. (original) The method of claim 25 further comprising the step of:

combining a previous state of the ACFs with the result of a condition code test specified by a current compare instruction to create a complex condition.

27. (original) The method of claim 25 wherein the condition code specifies a condition such as greater than (GT), less than (LT), equal (EQ) or less than or equal (LEQ).

28. (original) The method of claim 27 wherein the compare instruction is further utilized to specify the desired conditions to be tested and two source registers to be compared.

29. (original) The method of claim 28 wherein the compare instruction is further utilized to specify a data type covering packed data forms.

30. (original) The method of claim 28 wherein the compare instruction is further utilized to specify a Boolean combination specification field.

31. (original) The method of claim 26 further comprising the step of controlling branching in a sequence processor (SP) based upon the created complex condition.

32. (currently amended) The method of claim 26 further comprising the step of conditionally executing in a sequence processor (SP) and at least one processing element (PE) based on the created ~~couplex~~ complex consisting of a Boolean combination of multiple conditions, based upon the created complex condition.

33. (original) The method of claim 26 further comprising the step of conditionally executing on a combination of multiple conditions based upon the created complex condition.

Claims 34-55 (canceled).

56. (new) A method of conditional instruction execution comprising:

setting arithmetic scalar flags based on at least one side effect of the execution of a first instruction;

setting arithmetic condition flags based on the arithmetic scalar flags as specified by the first instruction;

determining whether to execute a second instruction based on the state of the arithmetic condition flags set by the first instruction; and

executing the second instruction if it is determined to execute the second instruction.

57. (new) The method of claim 56 wherein the first instruction is a compare instruction.

58. (new) The method of claim 57 wherein the at least one side effect is chosen from the group consisting of a carry indication, an overflow indication, a sign indication, and a zero indication.

59. (new) The method of claim 57 wherein the arithmetic scalar flags are set based on the execution of the compare instruction and a previously executed instruction.

60. (new) The method of claim 56 wherein the second instruction comprises at least one selectable conditional execution instruction opcode bit specifying the conditional execution of the second instruction.

61. (new) The method of claim 56 wherein the arithmetic condition flags are set based on a Boolean combination of the arithmetic scalar flags.

62. (new) The method of claim 56 wherein the execution of the second instruction affects the arithmetic scalar flags.

63. (new) The method of claim 56 wherein the first instruction performs an operation on packed data comprising a plurality of data elements, and wherein the method further comprises:

setting one arithmetic scalar flag for each data element of the packed data.

64. (new) The method of claim 56 wherein the execution of the second instruction affects the state of the arithmetic condition flags.

65. (new) The method of claim 56 wherein the first instruction is executed by a first processing element and the second instruction is conditionally executed by a second processing element.

66. (new) An apparatus for conditional instruction execution comprising:

means for setting arithmetic scalar flags based on at least one side effect of the execution of a first instruction;

means for setting arithmetic condition flags based on the arithmetic scalar flags as specified by the first instruction;

means for determining whether to execute a second instruction based on the state of the arithmetic condition flags set by the first instruction; and

means for executing the second instruction if it is determined to execute the second instruction.

67. (new) The apparatus of claim 66 wherein the first instruction is a compare instruction.

68. (new) The apparatus of claim 67 wherein the at least one side effect is chosen from the group consisting of a carry indication, an overflow indication, a sign indication, and a zero indication.

69. (new) The apparatus of claim 67 wherein the arithmetic scalar flags are set based on the execution of the compare instruction and a previously executed instruction.

70. (new) The apparatus of claim 66 wherein the second instruction comprises at least one selectable conditional execution instruction opcode bit specifying the conditional execution of the second instruction.

71. (new) The apparatus of claim 66 wherein the arithmetic condition flags are set based on a Boolean combination of the arithmetic scalar flags.

72. (new) The apparatus of claim 66 wherein the execution of the second instruction affects the arithmetic scalar flags.

73. (new) A processing apparatus for conditional instruction execution comprising:

a storage device for storing an arithmetic scalar flag;

an execution unit for executing a first instruction, generating an arithmetic scalar flag as a side effect of the execution, and storing the arithmetic scalar flag in the storage device;

a generation unit for generating an arithmetic condition flag utilizing both the stored arithmetic scalar flag and an opcode bit from the first instruction;

said execution unit for conditionally executing a second instruction based on the state of the arithmetic condition flag.

74. (new) The processing apparatus of claim 73 wherein the first instruction is a compare instruction.

75. (new) The processing apparatus of claim 74 wherein the at least one side effect is chosen from the group consisting of a carry indication, an overflow indication, a sign indication, and a zero indication.

76. (new) The processing apparatus of claim 74 wherein the arithmetic scalar flags are set based on the execution of the compare instruction and a previously executed instruction.

77. (new) The processing apparatus of claim 73 wherein the second instruction comprises at least one selectable conditional execution instruction opcode bit specifying the conditional execution of the second instruction.

78. (new) The processing apparatus of claim 73 wherein the arithmetic condition flags are set based on a Boolean combination of the arithmetic scalar flags.

79. (new) The processing apparatus of claim 73 wherein the execution of the second instruction affects the arithmetic scalar flags.